REMARKS

Affirmation of Election

Applicant hereby affirms the election without traverse of claims 1-24 and 26-41.

Objection to Drawings

The Examiner objected to the drawings "because the reference characters '550' and '540' have both been used to designate 540." A detailed analysis of the drawings shows they are correct and consistent. The reference designator 550 does not appear anywhere in the drawings. However, a detailed analysis of the specification revealed some typographical errors with respect to reference designators in the drawings. The specification has been amended herein to correct the reference designators in accordance with the drawings as originally filed. The drawings as originally filed are correct and comply with 37 C.F.R. §1.84(p)(4), so applicant respectfully requests reconsideration of the Examiner's objection to the drawings.

Request for Reference

In the office action, the Examiner requested a reference that discloses the prior art discussed in the patent application. The discussion of prior art is based on general knowledge in the art regarding the function of IBM products. Applicant is unaware of any particular reference that relates to the teachings discussed in the admitted prior art, and because applicant is under no duty to perform a search, applicant suggests that the Examiner simply rely upon the disclosure of the admitted prior art in applicant's disclosure.

Rejection of claims 1-4, 6-8, 13-16, 18-20, 26, 29-31, and 33-35 under 35 U.S.C. §102(a) as being anticipated by Taivalsaari

The Examiner rejected claims 1-4, 6-8, 13-16, 18-20, 26, 29-31, and 33-35 under 35 U.S.C. §102(a) as being anticipated by Taivalsaari. Each of these claims is addressed below.

Claim 13

Claim 13 recites:

> 13. A method for creating and enforcing protected system level Java code comprising the steps of:
>
> loading a plurality of Java classes, each of the plurality of Java classes that is protected including state data that indicates a protected class;
>
> performing a plurality of checks when each of the plurality of Java classes is loaded, the plurality of checks determining whether the class being loaded accesses at least one protected class, and if so, determining whether the class being loaded is authorized to access the at least one protected class, and generating an exception if the class being loaded is not authorized to access the at least one protected class.

In the rejection of claim 13, the Examiner cites to page 6 last paragraph and page 7 first paragraph of Taivalsaari as allegedly teaching the step in claim 13 of "performing a plurality of checks . . ." However, the Examiner has not shown any mapping of specific teachings of Taivalsaari on all of the limitations in claim 13. For this reason, the Examiner has failed to establish a prima facie case of anticipation for claim 13 under 35 U.S.C. §102(a).

Applicants respectfully assert that Taivalsaari does not teach all of the limitations in claim 13. Claim 13 expressly recites "performing a plurality of checks when each of the plurality of Java classes is loaded . . .". The discussion of access flags in Taivalsaari does not teach that any particular function is performed when each Java class is loaded. The plurality of checks that are performed when each class is loaded in claim 13 includes determining whether the class being loaded accesses at least one protected class. The access flags in Taivalsaari may indicate whether a particular class is protected or not, but there is no teaching or suggestion in Taivalsaari of determining whether a class being loaded accesses a different class that is protected. Claim 13 expressly involves multiple classes, *i.e.*, the class being loaded, and at least one protected class that the class being loaded may access. Nowhere does Taivalsaari teach or suggest that other classes are considered when a class is loaded.

Claim 13 further recites: "determining whether the class being loaded is authorized to access the at least one protected class". Again, Taivalsaari has no teaching whatsoever of determining whether a class being loaded is authorized to access a different class. The only teaching in Taivalsaari is that a class may include an AccessFlag object, which stores access flags such as "public", "protected", "private", "static", "native", "synchronized", etc. The fact that a class in Taivalsaari includes a "protected" or "private" flag does not mean that any checks are performed on the class when the class is loaded. Claim 13 expressly recites a plurality of checks that are performed when each class is loaded, which include determining whether the class being loaded is authorized to access one or more protected class(es) that is references. Nowhere does Taivalsaari teach or suggest determining whether a class being loaded accesses a protected class, and if so, performing other checks, as recited in claim 13.

The Examiner has taken a passing reference to AccessFlags in Taivalsaari and somehow assumes that based on this passing reference, Taivalsaari teaches the specific limitations in claim 13. In fact, the AccessFlags in Taivalsaari are considered such an unimportant part of the disclosure that they are not shown in FIG. 2 on page 6, and no further discuss of these AccessFlags is provided in Taivalsaari. The Examiner's position that Taivalsaari, based on this passing reference to AccessFlags, somehow teaches the detailed limitations in the claims is pure nonsense, and amounts to nothing more than speculation and wishful thinking on the part of the Examiner. Nowhere does Taivalsaari teach performing a plurality of checks when each of the plurality of Java classes is loaded. Nowhere does Taivalsaari teach that the plurality of checks that are performed when a class is loaded includes determining whether the class being loaded accesses at least one protected class. Nowhere does Taivalsaari teach that if the class being loaded accesses at least one protected class, determining whether the class being loaded is authorized to access the at least one protected class. Nowhere does Taivalsaari teach generating an exception if the class being loaded is not authorized to access the at least one protected class. In short, Taivalsaari teaches NONE of the limitations in the second clause of claim 13. For these reasons, claim 13 is allowable over Taivalsaari, and applicant respectfully requests reconsideration of the Examiner's rejection of claim 13 under 35 U.S.C. §102(a).

Claim 26

In rejecting claim 26, the Examiner reads the hashtables of Taivalsaari on the state/domain checker in claim 26. However, using hashtables to rapidly find methods and fields of a class at runtime has NOTHING whatsoever to do with determining whether a class being loaded accesses at least one protected class, as recited in claim 26. In addition, the execution requirements of a method cited in Taivalsaari do not determine whether a class being loaded is authorized to access the at least one protected class, as recited in claim 26. For these reasons, claim 26 is allowable over Taivalsaari, and applicant respectfully requests reconsideration of the Examiner's rejection of claim 26 under 35 U.S.C. §102(a).

Claim 1

Claim 1 was rejected based on the rejection of claim 13. However, claim 1 includes at least one processor and a memory that are not contained in claim 13. The Examiner has not addressed these limitations, and has therefore failed to establish a prima facie case of anticipation for claim 1 under 35 U.S.C. §102(a). Claim 1 contains limitations similar to claim 13 discussed in detail above, and is therefore allowable for the same reasons given above for claim 13. Applicant respectfully requests reconsideration of the Examiner's rejection of claim 1 under 35 U.S.C. §102(a).

Claims 2-4, 6-8, 14-16, 18-20, 29-31 and 33-35

Each of claims 2-4, 6-8, 14-16, 18-20, 29-31 and 33-35 depend on an independent claim that is allowable for the reasons given above. As a result, claims 2-4, 6-8, 14-16, 18-20, 29-31 and 33-35 are allowable as depending on allowable independent claims, and applicant respectfully requests reconsideration of the Examiner's rejection of these claims under 35 U.S.C. §102(a).

Rejection of claims 5, 17 and 32 under 35 U.S.C. §103(a) as being unpatentable over Taivalsaari in view of Bracha

The Examiner rejected claims 5, 17 and 32 under 35 U.S.C. §103(a) as being unpatentable over Taivalsaari in view of Bracha. Each of claims 5, 17 and 32 depend on an independent claim that is allowable for the reasons given above. As a result, claims 5, 17 and 32 are allowable as depending on allowable independent claims, and applicant respectfully requests reconsideration of the Examiner's rejection of these claims under 35 U.S.C. §103(a).

Rejection of claims 9-12, 21-24, 27, 28, and 36-41 under 35 U.S.C. §103(a) as being unpatentable over Taivalsaari in view of the admitted prior art (APA)

The Examiner rejected claims 9-12, 21-24, 27, 28, and 36-41 under 35 U.S.C. §103(a) as being unpatentable over Taivalsaari in view of the admitted prior art (APA). Each of these claims is addressed below.

Claim 39

In rejecting claim 39, the Examiner reads the hashtables in Taivalsaari on the state/domain checker in claim 39. The hashtables in Taivalsaari are used "for rapidly finding the various methods and fields of the class at runtime." Nowhere does Taivalsaari teach or suggest that these hashtables perform ANY kind of checks within the scope of claim 39.

Claim 39 recites five different checks that are performed by the state/domain checker. Check C1 is performed during class verification. Check C2 is performed during class preparation. Check C3 is performed during class resolution. Checks C4 and C5 are performed at runtime. Nowhere does Taivalsaari teach or suggest that different checks may be performed during these different stages.

Check C1 recites a catalog of allowed classes that is generated during a JVM build process. Nowhere does Taivalsaari teach or suggest such a catalog of allowed classes, so it cannot teach the check in C1 to determine whether the class being verified is included in the catalog of allowed classes.

Check C2 determines whether a class being prepared has a superclass. The Examiner reads the Class ConstantPoolEntry on p. 6 of Taivalsaari on the superclass in claim 39. The ConstantPoolEntry class in Taivalsaari is an abstract class that has a number of concrete subclasses. This means that ConstantPoolEntry is a superclass. However, there is no teaching in Taivalsaari of performing a check to see if a concrete subclass has a superclass, and if so, performing the check at lines 14-17 of claim 39. Check C2 requires that the class being prepared implement at least the same private domain interface or system state interface as the superclass, otherwise an exception is thrown. Taivalsaari is devoid of any teaching of a private domain interface or system state interface, so it cannot teach these limitations in check C2.

Applicants could go on and show detailed reasons why neither Taivalsaari nor the APA teach or suggest the remaining checks C3-C5 in claim 39. However, it is abundantly clear from the discussion of checks C1 and C2 above that Taivalsaari does not teach these limitations. As a result, claim 39 is allowable over the combination of Taivalsaari and APA, and respectfully request reconsideration of the Examiner's rejection of claim 39 under 35 U.S.C. §103(a).

Claim 12

Claim 12 was rejected based on the rejection of claim 39. However, claim 12 includes at least one processor and a memory that are not contained in claim 13. The Examiner has not addressed these limitations, and has therefore failed to establish a prima facie case of anticipation for claim 12 under 35 U.S.C. §102(a). Claim 12 contains limitations similar to claim 39 discussed in detail above, and is therefore allowable for the same reasons given above for claim 39. Applicant respectfully requests reconsideration of the Examiner's rejection of claim 12 under 35 U.S.C. §103(a).

Claim 24

Claim 24 was rejected based on the rejection of claim 39. Claim 39, however, is allowable as indicated above. As a result, claim 24 is likewise allowable, and applicant respectfully requests reconsideration of the Examiner's rejection of claim 24 under 35 U.S.C. §103(a).

Claims 9-11, 21-23, 27, 28, 36-38 and 40-41

Claims 9-11, 21-23, 27, 28, 36-38 and 40-41 each depend on an independent claim that is allowable for the reasons given above. As a result, claims 9-11, 21-23, 27, 28, 36-38 and 40-41 are allowable as depending on allowable independent claims.

Invitation to the Examiner

The Examiner's rejection of the pending claims fails to address many of the limitations in those claims. In particular, Taivalsaari has no teaching or suggestion of different checks at different times. The claims specifically recite when certain checks are performed (*e.g.*, when a class is loaded, at runtime, during class verification, during class preparation, during class resolution). Should the Examiner decide to maintain any of the pending rejections, applicants respectfully invite the Examiner to specifically map the teachings of the cited art to each and every limitation in the claims, including when the checks are performed, so the Examiner's rejections may be adequately addressed on appeal.

Conclusion

In summary, none of the cited prior art, either alone or in combination, teach, support, or suggest the unique combination of features in applicant's claims presently on file. Therefore, applicant respectfully asserts that all of applicant's claims are allowable. Such allowance at an early date is respectfully requested. The Examiner is invited to telephone the undersigned if this would in any way advance the prosecution of this case.

Respectfully submitted,

By__________________________
Derek P. Martin
Reg. No. 36,595

MARTIN & ASSOCIATES, L.L.C.
P.O. Box 548
Carthage, MO 64836-0548
(417) 358-4700